

**In the Claims**

The following Listing of Claims replaces all prior versions in the application:

LISTING OF CLAIMS

1. (Previously Presented) A method for controlling communications in a system for automatically distributing a software update to a network of devices controlled by an organization, the method comprising:  
receiving a first request for available updates from a user interface, the request received by a web module;  
processing the first request on the web module;  
sending a second request for available updates from the web module to a main module;  
placing the web module in a listen state without waiting for a response to the second request;  
processing the second request on the main module;  
sending a third request for available updates from the main module to a patch module;  
placing the main module in a listen state without waiting for a response to the third request;  
processing the third request on the patch module, the processing including examining a global update repository including updates from multiple application manufacturers;  
sending a first reply with a list of available updates from the patch module to the main module;  
sending a second reply with a list of available updates from the main module to the web module; and  
sending a third reply with a list of available updates from the web module to the user

interface.

2. (Previously Presented) The method of claim 1, wherein the web module is in a listen state, process state, or respond state.
3. (Previously Presented) The method of claim 1, wherein the main module is in a listen state, process state, or respond state.
4. (Previously Presented) The method of claim 1, wherein the patch module is in a listen state, process state, or respond state.
5. (Previously Presented) The method of claim 1, wherein the web module, main module, and patch module are located on a client.
6. (Previously Presented) The method of claim 1, wherein the web module, main module, and patch module are located on a server.
7. (Previously Presented) The method of claim 5, wherein the patch module communicates with a patch module on a server.
8. (Previously Presented) The method of claim 6, wherein the patch module communicates with a patch module on a client.
9. (Previously Presented) The method of claim 1, wherein each of the requests is an Extensible

Markup Language (XML) schema.

10. (Previously Presented) The method of claim 1, wherein each of the replies in an Extensible Markup Language (XML) schema.
11. (Previously Presented) The method of claim 2, wherein in the listen state, a module waits for communication with another module.
12. (Previously Presented) The method of claim 2, wherein in the process state, a module:  
determines the grammatical correctness of a request; and  
generates a request.
13. (Previously Presented) The method of claim 2, wherein in the respond state, a module  
further generates a reply message.
14. (Previously Presented) The method of claim 3, wherein in the listen state, a module waits for communication with another module.
15. (Previously Presented) The method of claim 3, wherein in the process state, a module:  
determines the grammatical correctness of a request; and  
generates a request.
16. (Previously Presented) The method of claim 3, wherein in the respond state, a module  
further generates a reply message.

17. (Previously Presented) The method of claim 1, further comprising:

validating the requests for syntactical correctness upon receipt by a module.

18. (Currently Amended) An apparatus for controlling communications in a system for automatically distributing a software update to a network of devices controlled by an organization, the apparatus comprising:

a memory;

a web module user interface request receiver;

a web module user interface request processor coupled to the web module user interface request receiver;

a web module request sender coupled to the web module user interface request processor;

a web module listen state placer coupled to the web module request sender and configured to place the web module in a listen state without waiting for a response to a request sent by the web module request sender;

a main module web module request processor coupled to the web module request sender;

a main module request sender coupled to the main module web module request processor;

a main module listen state placer coupled to the main module request processor configured to place the main module in a listen state without waiting for a response to a request sent by the main module request sender;

a patch module main module request processor coupled to the main module request sender and configured to examine ~~examine~~ a global update repository including updates from multiple application manufacturers;

a patch module reply sender coupled to the patch module main module request processor;

a main module reply sender coupled to the patch module reply sender; and a web module  
reply replay sender coupled to the main module reply sender.

19. (Original) The apparatus of claim 18, wherein the apparatus is located on a server.
20. (Original) The apparatus of claim 18, wherein the apparatus is located on a client.
21. (Currently Amended) An apparatus for controlling communications in a system for  
automatically distributing a software update to a network of devices controlled by an  
organization, the apparatus comprising:
- a memory;
- means for receiving a first request for available updates from a user interface, the request  
received by a web module;
- means for processing the first request on the s-aid web module;
- means for sending a second request for available updates from the web module to a main  
module;
- means for placing the web module in a listen state without waiting for a response to the  
second request;
- means for processing the second request on the main module;
- means for sending a third request for available updates from the main module to a patch  
module;
- means for placing the main module in a listen state without waiting for a response to the  
third request;
- means for processing the third request on the patch module, the processing including

examining a global update repository including updates from multiple application manufacturers;

means for sending a first reply with a list of available updates from the patch module to the main module;

means for sending a second reply with a list of available updates from the main module to the web module; and

means for sending a third reply with a list of available updates from the web module to the user interface.

22. (Previously Presented) The apparatus of claim 21, wherein the web module is in a listen state, process state, or respond state.
23. (Previously Presented) The apparatus of claim 21, wherein the main module is in a listen state, process state, or respond state.
24. (Previously Presented) The apparatus of claim 21, wherein the patch module is in a listen state, process state, or respond state.
25. (Previously Presented) The apparatus of claim 21, wherein the web module, main module, and patch module are located on a client.
26. (Previously Presented) The apparatus of claim 21, wherein the web module, main module, and patch module are located on a server.

27. (Previously Presented) The apparatus of claim 25, wherein the patch module communicates with a patch module on a server.
28. (Previously Presented) The apparatus of claim 26, wherein the patch module communicates with a patch module on a client.
29. (Previously Presented) The apparatus of claim 21, wherein each of the requests is an Extensible Markup Language (XML) schema.
30. (Previously Presented) The apparatus of claim 21, wherein each of the replies is an Extensible Markup Language (XML) schema.
31. (Previously Presented) The apparatus of claim 22, wherein in the listen state, a module waits for communication with another module.
32. (Previously Presented) The apparatus of claim 22, wherein in the process state, a module:  
determines the grammatical correctness of a request; and  
generates a request.
33. (Previously Presented) The apparatus of claim 22, wherein in the respond state, a module further generates a reply message.
34. (Previously Presented) The apparatus of claim 23, wherein in the listen state, a module waits for communication with another module.

35. (Previously Presented) The apparatus of claim 23, wherein in the process state, a module:  
determines the grammatical correctness of a request; and generates a request.
36. (Previously Presented) The apparatus of claim 23, wherein in the respond state, a module  
further generates a reply message.
37. (Previously Presented) The apparatus of claim 21, further comprising:  
Validating the requests for syntactical correctness upon receipt by a module.
38. (Previously Presented) A program storage device readable by a machine, tangibly  
embodying a program of instructions executable by the machine to perform a method for  
controlling communications in a system for automatically distributing a software update to a  
network of devices controlled by an organization, the method comprising:  
receiving a first request for available updates from a user interface, the request received by a  
web module;  
processing the first request on the web module;  
sending a second request for available updates from the web module to a main module;  
placing the web module in a listen state without waiting for a response to the second request;  
processing the second request on the main module;  
sending a third request for available updates from the main module to a patch module;  
placing the main module in a listen state without waiting for a response to the third request;  
processing the third request on the patch module, the processing including examining a  
global update repository including updates from multiple application manufacturers;



sending a first reply with a list of available updates from the patch module to the main module;

sending a second reply with a list of available updates from the main module to the web module; and

sending a third reply with a list of available updates from the web module to the user interface.